Chapter 3
Intermediate CSS

- / font-family */*

Decides the font type. Type name must be between quotes if it starts with numbers.

```
h1 {
   font-family: Arial;
}
```

- / font-size */*

Here 'em' is dynamically changing and multiplying the text that it belongs to, according to the one level outer text of the current text

```
body {
   font-size: 16px;
}

h1 {
   font-size: 2.0em;
}
```

- / font-weight */*

Decides how bold/light text will be. Some of the font types let you use it with a range of 100-800 (not every value but every hundred value) other than normal, bold etc.

```
p {
   font-weight: 800;
}
h2 {
   font-weight: bold;
}
```

- / line-height */*

Its value multiplies the default line spacing value which is determined by the font

```
p {
   line-height: 2;
}
```

- / text-align */*

Specifies the alignment

```
h1 {
   text-align: center;
}
```

- / text-decoration */*

Used to give text underline effect, linethrough effect etc.

```
p {
   text-decoration: line-through;
}
```

- fonts.google.com

Free fonts that Google provides us; for usage:
https://www.youtube.com/watch?v=Z3JR6mEWEEo&feature=youtu.be

- Box Model
    In a document, each element is represented as a rectangular box. In CSS, each of these rectangular boxes is described using the standart box model.
    Each box has four edges:
    1.margin edge: the space outside the box; kinda within the elements. Usages,
    ```
    margin: 20px; /* Same space from all sides */
    margin-top: 40px;
    margin: TOP RIGHT BOTTOM LEFT; /* More specified space from any side */
    margin: 0 auto 0 auto; /* Centers the element from left and right edges */
    margin: 0 auto; /* Specifies top to 0 px & right to auto, so indirectly fully centering from right
    ```
and left again */

    2.border edge: the thickness of the border. Usages,
    ```
    border: 2px solid blue;
    border-width: 2px;
    border-style: solid;
    border-color: blue;
    ```

    3.padding edge: the space between border and the content inside of it. Usages,
    ```
    padding: 10px;
    padding-left: 20px;
    ```

    4.content edge: the text itself (width & hight). Usages,
    ```
    width: 20px;
    width: 60%;
    ```

    The space on the right of the content comes from the number 4, which is content edge -width- but the space on the left comes from number 3, which is padding edge -padding-left-. On the other hand, padding has also right spacing, which means we have two adjustments for the right side of our content.

    We can use a percentage on the width to make the website more dynamic and change itself depending on the body (window) size,
    ```
    p {
        width: 50%;
    }
    ```

- An element can have multiple classes at once,
    ```
    <td class="vertical horizontal"></td>
    ```
    So this td is in also vertical class and also horizontal class.

- Normally each div and div-like elements go on different lines, that's because what's defined as default as float property. If we change it as the following, all divs will float from left to right to bottom:
    ```
    div {
        float: left
    }
    ```
    When we don't modify float property, by default HTML has one space between adjacent images; but when we modify it, that's gone.

- To go to end and the beginning of a line in Sublime with "Alt+Right"/"Alt+Left"; paste the following into the Key Bindings within Preferences:
    ```
    { "keys": ["alt+left"], "command": "move_to", "args": {"to": "bol", "extend":
    false} },
    { "keys": ["alt+right"], "command": "move_to", "args": {"to": "eol",
    "extend": false} }
    ```

- It may be more beneficial to have just one h1 -which is the main and the only one- on the page and having the other headings -which are multiple- as h2.

- 'rem' adjusts the text size not according to the size of the parent element like 'em' but according to the root element; which means we don't have to worry about one changing the other,

```
letter-spacing: 0.2rem;
```
Letter spacing -btw- specifies the space between letters

- 'max-width' specifies the max width to be shown as window shrinkes. It makes sense to use it with a percentage width.

```
max-width: 800px;
width: 80%;
```

- `<hr>` tag adds a horizontal line on the page. No closing tags etc.